

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (Previously Presented) A system for analyzing thread deadlocks in a Java virtual machine, comprising:
  - a thread analyzer tool, wherein the tool analyzes a thread dump to automatically identify thread deadlocks, wherein the tool identifies threads that are in a self wait condition and threads that are in a circular wait condition; and
  - a user interface that allows a user to specify criteria, wherein the tool excludes threads that do not meet the criteria from being identified on the user interface as deadlocked threads even though the threads were identified as being deadlocked threads pursuant to the thread dump analysis.
2. (Cancelled)
3. (Original) The system of claim 1, wherein the tool analyzes a thread dump file in offline mode.
4. (Original) The system of claim 1, wherein the tool obtains a thread dump from a running information processing system.
5. (Previously Presented) A method of analyzing thread deadlocks in a Java virtual machine, comprising the steps of:
  - obtaining a thread dump of a Java virtual machine;
  - analyzing the thread dump to automatically identify threads in a deadlock condition, wherein threads in a circular wait condition and threads in a self wait condition are identified; and
  - receiving, by a user interface, user specified criteria from a user, wherein identified deadlock threads that do not meet the criteria are filtered such that they are not presented to the user by the user interface even though the threads that do not meet the criteria were identified as being deadlocked threads pursuant to the thread dump analysis.
6. (Cancelled)

7. (Previously Presented) The method of claim 5, wherein filtered threads are excluded from identification as threads in a self wait condition.
8. (Original) The method of claim 5, wherein the tool analyzes a thread dump file in offline mode.
9. (Previously Presented) The method of claim 5, wherein a matrix is populated to identify threads owning resources and threads waiting on resources, and wherein the matrix is used to identify threads in a circular wait condition.
10. (Previously Presented) The method of claim 5, wherein the step of analyzing the thread dump to automatically identify threads in a deadlock condition includes the steps of:
  - (a) identifying, in the thread dump, a locked object that is already in use by a thread;
  - (b) identifying threads that own resources the thread that has locked the locked object;
  - (c) identifying each waiting thread that is waiting on the locked object; and
  - (d) comparing the results from steps (b) and (c), and then repeating steps (a)-(c) for each locked object in the thread dump, to identify the threads in the circular wait condition and the self wait condition.
- 11-22. (Cancelled)